**Software Requirement Specification (SRS) for Online Voting System(OVT)**

**1. Introduction**

1.1 **Purpose**:

Online Voting System (OVS) is intended to provide an easier, more convenient alternative to the tedious costly offline voting process of an election. It provides a platform for voters to cast votes online, election commission to approve candidates based on their documents, and candidates to share their profiles and accomplishments with their constituency. With automated vote counting, results declaration, and statistics generation, this software aims to completely replace the traditional election procedure.

This document is meant to delineate the features of OVS, so as to serve as a guide to the developers on one hand and a software validation document for the prospective client on the other

**1.2 Scope:**

We describe what features are in the scope of the software and what are not in the scope of the software to be developed.

In Scope:

a) Authentication of users (candidate/voter/admin).

b) Registration of electoral candidates and voters.

c) Verification of candidate profiles.

d) Profile management of electoral candidates, which includes uploading personal details and milestones.

e) Provision for elections and automated vote counting.

f) Managing elections and finding winners from each area.

g) Cancelling or pausing elections in case of security breaches or violation of other laws as decided by the Election Commission.

h) Generation of statistics after election.

Out of Scope:

a) Any election related prediction.

b) Evaluation of candidate profiles.

c) Automatic verification of uploaded documents.

**1.3 Definitions, Acronyms, and Abbreviations:**

*Acronyms and Abbreviations:*

a) OVS: Online Voting System.

b) EC: Election Commission.

c) SRS: Software Requirements Specification.

*Definitions:*

1. Election : A formal and organized choice by vote of a person for a political office or other position.
2. Electoral Commission: An independent body set up by the Indian Parliament for the purpose of organizing and monitoring elections.
3. Voter: An Indian citizen above the age of eighteen, who has registered himself/herself with the Electoral Commission for the purpose of voting.
4. Candidate: An Indian citizen above the age of thirty, who has voted before, and wishes to represent his constituency in an election.

**1.4 Overview:**

The rest of this SRS is organized as follows: Section 2 gives an overall description of the software. It gives what level of proficiency is expected of the user, some general constraints while making the software and some assumptions and dependencies that are assumed. Section 3 gives specific requirements which the software is expected to deliver. Functional requirements are given by various use cases. Some performance requirements and design constraints are also given

**2. Overall Description:**

**2.1 Product Perspective:**

OVS is aimed towards providing an online platform for conducting elections in an efficient and cost-effective way. OVS should be user-friendly, easy to use, and reliable for the mentioned purpose.

OVS is intended to be a stand-alone web-based application and should not depend on the availability of other software other than a modern web browser. It should run on UNIX, Windows and IOS based platforms.

**2.2 Product Functions:**

**2.3 User Characteristics:**

a) User should have sufficient expertise to use a web-based application.

b) All voters and candidates must possess a valid Voter ID card.

**2.4 Principal Actors:**

The principal actors in OVS are “admin”, “voter”, “candidate”, and the “system”

**2.5 General Constraints:**

a) For working, OVS requires an internet connection.

b) OVS is a web-based software.

**2.6 Assumptions and Dependencies:**

a) Full working of OVS is dependent on the availability of an internet connection.

b) OVS can only be used to vote in selected areas.

**3 Specific Requirements:**

**3.1 Functional Requirements:**

We describe the functional requirements by giving various use cases.

***Use cases related to system authorization***

**Use Case 1**: User Login

*Primary Actor*: User

*Pre Condition*: Nil

*Main Scenario* :

1. Start the application. User prompted for login and password.

2. User gives the login and password.

3. System does authentication.

4. The user dashboard is displayed according to the category of user(admin/voter/candidate).

*Alternate Scenario* :

4(a). Authorization fails

4(a)1. Prompt the user that he typed the wrong password

4(a)2. Allow him to re-enter the password. Give him 3 chances.

**Use Case 2:** Change Password

*Primary Actor*: User

*Pre Condition*: User logged in

*Main Scenario*:

1. User initiates the password change command.

2. User is prompted for old password, new password and confirm new password.

3. User gives the old password, new password and confirm new password.

4. System does authentication.

5. New password is registered with the system.

*Alternate Scenario*:

4(a). Authorization fails

4(a)1. Prompt the user that he typed the wrong password

4(a)2. Allow him to re-enter the password. Give him 3 chances.

4(b). New password and confirm new password do not match.

4(b)1. Allow him to re-enter the attributes. Give 3 chances.

**Use Case 3: Candidate Document Verification**

*Primary Actor*: Admin

*Pre Condition*: Admin logged in.

*Main Scenario*:

1. Admin clicks a candidate name marked with the status “pending approval”.
2. Admin checks the documents of the candidate.
3. If the document is correct, he clicks to Approve button. If it is faulty, admin can click on reject button to. Admin can also remove faulty document by using Remove button.
4. Admin clicks on “register Candidate”. This generates user id and password for candidate.

*Alternate Scenario*:

4(a). All documents present do not have the status “Approved”

4(a)1. System asks the admin to delete the faulty document or wait for the candidate to upload the correct one.

4(b) A document has not been marked as Approve/Reject.

4(b)1. System prompts the admin to mark it.

**Use Case 5:** Voters view Candidate’s data

*Primary Actor*: Voter

*Pre-Condition*: Voter logged in.

*Main Scenario*:

1. Voter enters the area in which he wants to view candidates.

2. He clicks on the desired candidate name.

3. System displays the documents and detailed information about the background of candidate.

*Alternate Scenario*:

2(a). No candidates exist from that area

4(a)1. System displays apology message and allows user to enter another area.

**Use Case 6:** Upload Candidate Information

*Primary Actor*: Candidate

*Pre-Condition*: Candidate logged in.

*Main Scenario*:

1. Candidate clicks on add document button.

2. System asks candidate to choose the directory containing the document.

3. Candidate adds the document that may include previous milestones or other details.

*Alternate Scenario*:

*4(a)* File format not supported

4(a)1. System displays an apology message and allows candidate to choose an alternative document.

4(b) File size exceeded

4(b)1. System displays an apology message and allows user to delete uploaded documents.

***Use Cases related to registration***

**Use Case 7:** Candidate Registration

*Primary Actor*: Candidate

*Pre-Condition*: Nil

*Main Scenario* :

1. System asks candidate to enter basic details like Name, Age, Party, Constitution etc.

2. Candidate enters the details.

3. System asks candidate to upload the requisite documents.

4. Candidates browses to the directory containing the documents and uploads them.

5. Candidate clicks on Register button.

6. System notifies that the user id and password will be given after verification of the documents.

*Alternate Scenario*:

3(a) Candidate’s age < 30 years

3(a)1. System shows an apology message that the user can’t be a candidate.

3(b) Candidate enter invalid constitution

3(b)1. System displays error message

3(c) Candidate left a field blank

3(b)1. System displays error message and allows candidate to type the information

5(a) File limit exceeded

5(a)1. System displays an apology message and allows user to delete uploaded documents.

6(a) Candidate hasn’t uploaded any documents

6(a)1. System displays an error message.

**Use Case 8:** Voter Registration

*Primary Actor*: Voter

*Pre-Condition*: Nil.

*Main Scenario*:

1. System asks candidate to enter basic details like Name, Age, Voter Card ID, Email ID etc.

2. Voter enters the details.

3. Voter clicks on Register button.

4. System registers the voter and displays the auto generated user id and password.

*Alternate Scenario*:

3(a) Voter’s age < 18 years

3(a)1. System shows an apology message that the user can’t be a voter.

5(a) Voter left a field blank

3(b)1. System displays error message and allows voter to type the information

**Use Case 9:** Vote for Candidate

*Primary Actor*: Voter

*Pre-Condition*: Voter logged in.

*Main Scenario* :

1. System displays the different candidates with their details.
2. Voter clicks on the vote button beside the desired candidate.
3. System displays a success message and disables the voter from voting further.

*Alternate Scenario*

*Use cases related to transactions:*

**Use Case 10:** Result Calculation

*Primary Actor*: Admin.

*Pre-Condition*: admin logged in.

*Main Scenario*:

1. Admin clicks on “finish election” button.

2. Admin click on “generate result”.

3. System displays the candidates vs their votes and the winner.

4. System updates dashboard of all users with the results.

*Alternate Scenario:*

3(a). Election is ongoing.

3(a)1. Systems displays message that result can’t be generated while the election is ongoing.

**Use Case 11:** Election Creation

*Primary Actor*: Admin.

*Pre-Condition*: Admin logged in.

*Main Scenario* :

1. Admin clicks on “Create Election”.
2. System displays the list of registered candidates.
3. Admin chooses/approves the candidates participating in the election.
4. Admin clicks on “Create”
5. System displays a success message.

*Alternate Scenario:*

4(a). Admin hasn’t approved any candidate

4(a)1. System displays an error message

4(b) Another election is ongoing

4(b)1. System displays an error message.

2(a) There are no registered candidates.

2(a)1. System displays an apology message and doesn’t create the election.

**3.2 Performance Requirements:**

(a) The software should be able to log in and feed the voter with new pages on request with a response time of the order of a few seconds.

(b) The software is expected to serve a maximum of up to 500 voters at any point of time.

(c) The software should have the capability for processing about 20 transactions each second.

(d) Should run on at least 1.0 GHz, 512 MB machine.

**3.3 Design Constraints:**

1. *Security:* The files containing the information regarding votes and candidates should be secured against malicious deformations.

2. *Fault Tolerance*: Data should not become corrupted in case of system crash or power failure.

**3.4 Safety Requirements:**

(a) In order to prevent data loss in case of power failure, the result of votes that were polled till then have to be saved in the database, for the system to resume the counting process on reboot

(b) In case the EC detects any security lapse in the system, they should able to pause the election immediately while preserving the already polled votes.

(c) The software should be capable of recovering from crashes and continuing the voting process.

**3.5 Security Requirements:**

(a) The system should provide basic security features like password authentication and encrypted transactions.

(b) All the passwords generated and communicated to the users should be stored in the server only in an encrypted form for login management to prevent misuse.

(c) Serial attacks should be avoided by maintaining a minimum time gap between successive invalid log-in attempts.

(d) Additional security features like voter anonymity should be provided.